









;` fcbVgUf[a`

Ea\_ WZ[` Yiel cb` Y

---

## ? k 3bbcbSLZ

---

The focus of *Cracking the Coding Interview* is algorithm, coding and design questions. Why? Because while you can and will be asked behavioral questions, the answers will be as varied as your resume. Likewise, while many firms will ask so-called “trivia” questions (e.g., “What is a virtual function?”), the skills developed through practicing these questions are limited to very specific bits of knowledge. The book will briefly touch on some of these questions, to

---







Amazon's recruiting process usually begins with one or two phone screens in which you interview with a specific team. The engineer who interviews you will usually ask you to write simple code and read it aloud on the phone. They will ask a broad set of questions to explore what areas of technology you're familiar with.

Next, you









.....  
.....





























# The Interview and Beyond





















.....

.....



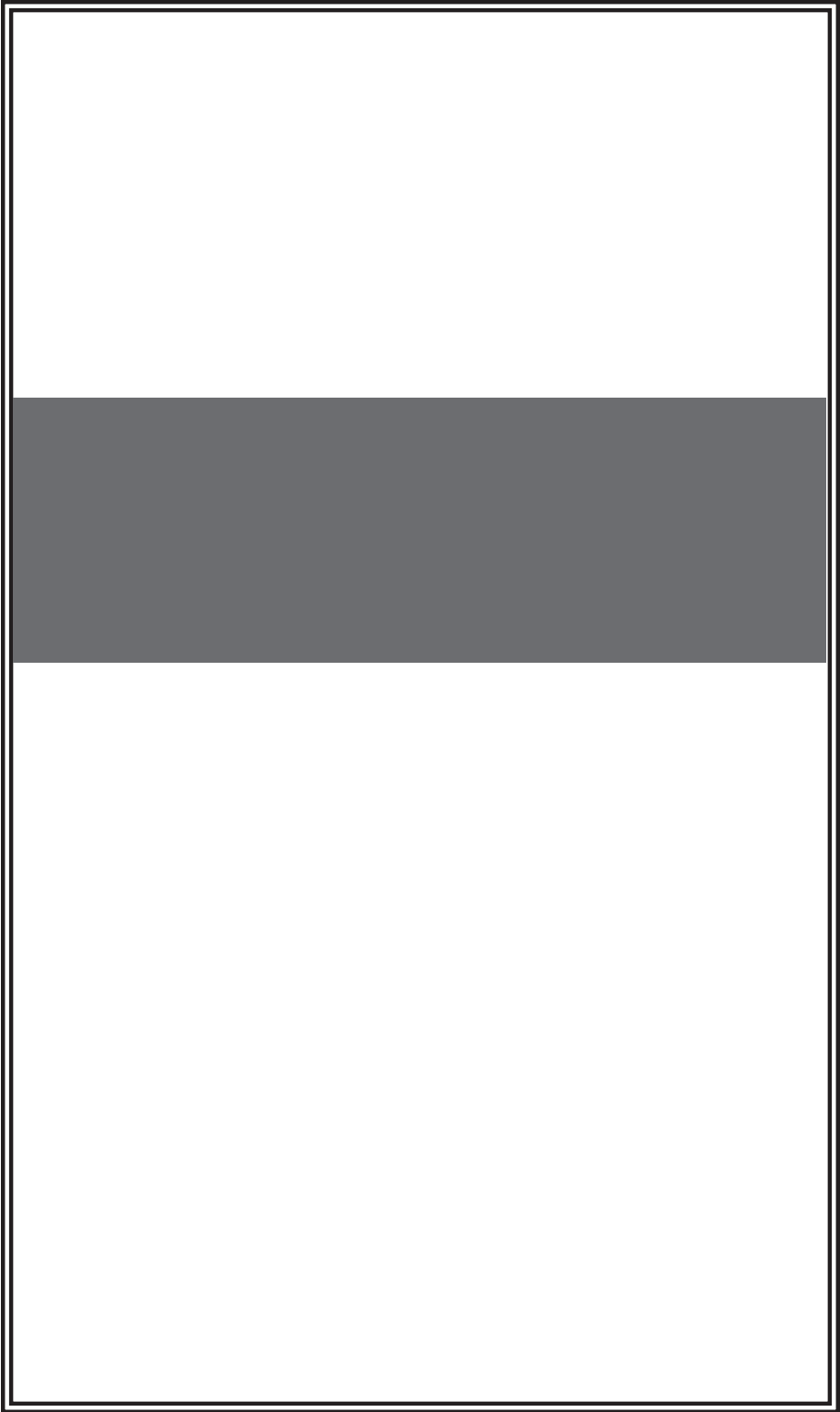












*Part 1*  
Data Structures

















8z#

*Part 2*  
Concepts and Algorithms













)# Design the data structures for a generic deck of cards. Explain how you would subclass it to implement particular card games.

pg 151

)\$ Imagine you have a call center with three levels of employees: fresher, technical lead (TL), product manager (PM). There can be multiple employees, but only one TL or PM. An incoming telephone call must be allocated to a fresher who is free

c(A)the ca(v) 10(, )8-10h(ee)8-10 (or )8-10(eser)8-10 (must )80(es (coca) 4 (t) 6 er)8-10 (the)8-10























*Part 3*  
KnowledgeBased



#%#

















#)







*Part 4*  
Additional Review Problems





\$"z#

Input: DAMP, LIKE

Output: DAMP -> LAMP -> LIMP -> LIME -> LIKE

pg 291

Each problem may have many 'optimal' solutions that differ in runtime, space, clarity, extensibility, etc. We have provided one (or more) optimal solutions. If you have additional solutions you would like to contribute, please contact us at <http://www.xrl.us/ccbook> or [support@careercup.com](mailto:support@careercup.com).

We welcome all feedback and suggestions. Contact us at <http://www.xrl>



# Solutions











#2 Write a method to replace all spaces in a string with '%20':

pg 48

## Easy to Learn

---

The algorithm is as follows:

1. Count the number of spaces during the first scan of the string.
2. Parse the string again from the end and for each character:
  - »







Ea^gf[a` efa 5ZSbfVd#

1.6 Write code to remove duplicates from an unsorted linked list

---





- Example** You have two numbers represented by a linked list, where each node contains a single digit. The digits are stored in reverse order, such that the 1's digit is at the head of the list.
-







Describe how you could use a single

---





















%

---









86 Given a node in a binary tree, write an algorithm to find the 'next' node (e.g.

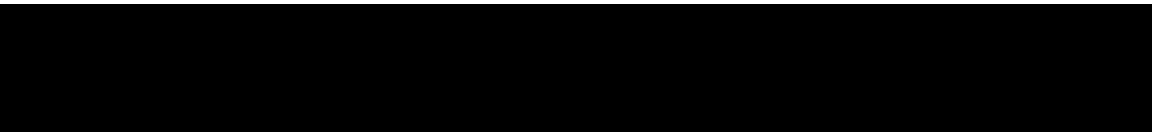
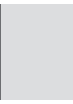
---



84

---























- 2 Write a function to determine the number of bits required to convert integer A to integer B.

Input: 31, 14

---





-





(Z

---

Q1. You have a

---

---



»

Ea'gf[a` efa 5ZSbfVd(











)

---

Ea^gf[a` efa 5ZSbfVd)



2 Design the data structures for an online book reader system.

pg 62

**EA>GF;A@**

---

Since the problem doesn't describe much about the functionality, let's assume we want to







































Ea^gf[a` efa 5ZSbfVd\*

\*2) Given an infinite number of quarters (25 cents), dimes (10 cents), nickels (5 cents) and pennies (1 cent), write code to calculate the number of ways of representing n cents.

pg 64

## EA>GF;A@

---

This is a recursive problem, so let's figure out how to do makeChange(n) using prior solutions (i





Ea^gf[a` efa 5ZSbfVd+















Z} } yy}C6' &)-({/, -46)}(6\*&6-86  
Y





















}





Ea^gf[a` efa5ZSbfVd##









Ea^gf[a` efa5ZSbfVd##





















Ea^gf[a` efa 5ZSbfVd#\$ | Testing















#%¿ What is name hiding in C++?

pg 76

**EA>GF;A@**

---

Let us explain through an example.















Q: What is the difference between `final`, `finally`, and `finalize`?

pg 78

**EA>GF;A@E**

---

*Final*

When applied to a variable (



Ea^gf[a` efa 5ZSbfVd#&| Java

#&Z

---









# 2%

---





Explain the following terms: virtual memory, page fault, thrashing

---

#(Z\$

---



#(z& Write a step by step execution of things that happen after a user presses a key on the keyboard. Use as much detail as possible.

pg 82

EA>GF;A@

---









#(ž

---



3 ns after a read begins. Uninteresting time segments are surrounded by [brackets]. Each character represents 1 ns









Q1 Explain any common routing protocol in detail. For example: BGP, OSPF, RIP

---

- » Startup.











Q: What's the difference between a thread and a process?

pg 86

## EA>GF;A@

---

Processes and threads are related to each other but are fundamentally different.

A process can be thought of as an instance of a program in execution. Each process is an independent entity to which system resources (CPU time, memory, etc.











#\*







Ea^gf[a` efa 5ZSbfVd#+ | Moderate





Ea^gf[a` efa 5ZSbfVd#+ | Moderate







Ea^gf[a` efa 5ZSbfVd#+ | Moderate















Design an algorithm to

---

---

Ea^gf[a` efa 5ZSbfVd\$" | Hard



Earliest valid time | Hard









Ea^gf[a` efa 5ZSbfVd\$" | Hard







Ea^gf[a` efa 5ZSbfVd\$" | Hard







^X} } } }



```
^] } } } } r  
^\\ } } } } (?!6ccs  
^[ ] } } } } r  
^Z } } } } r  
^Y } } } } 8, / ``s  
^X } } } } r  
^W } } } } )6' &)-}(*s  
]V } r  
]_ }  
]^ }
```

Ea^gf[a` efa 5ZSbfVd\$" | Hard







Given a dictionary of millions of words, give an algorithm to find the largest possible rectangle of letters such that every row forms a word.

---

Earliest valid time | Hard





